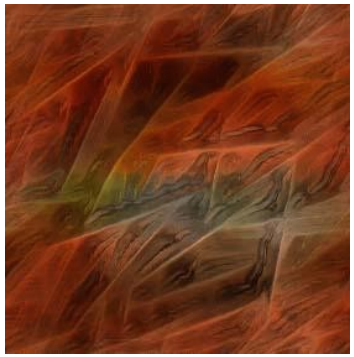


Application of Templates and Metrics to Enhance and Assess Systems Engineering Effectiveness in the IT Sector

Dr. Dinesh Verma

Associate Dean and Professor, Stevens Institute of Technology

Mr. Paul Popick

Senior Systems Engineering Manager, IBM Corporation

Presentation Outline

- Relevance of Systems Engineering to the IT Industry
  - Forensics of a project close to home…
- Adapted Systems Engineering Process for the IT Industry – The IBM Approach
  - Baselines and Reviews
  - Sample Templates
- Cost of Systems Engineering
  - Sample Staffing Structures
- Benefits of Systems Engineering
  - Difficulties with metrics
- Concluding Remarks

Presentation Outline

- Relevance of Systems Engineering to the IT Industry
  - Forensics of a project close to home…
- Adapted Systems Engineering Process for the IT Industry – The IBM Approach
  - Baselines and Reviews
  - Sample Templates
- Cost of Systems Engineering
  - Sample Staffing Structures
- Benefits of Systems Engineering
  - Difficulties with metrics
- Concluding Remarks

System Development and Integration challenges in the information technology sector.

UK Ministry of Defense	UK Civil Information Technology	US Civil Information Technology
	10-20% met success criteria	16% project success
Top 25 programs slipped 35-40 mos. on average	40-50% late, over-budget, or did not meet technical goals	53% project challenged
10% of projects missed key technical requirements	40% failed or were abandoned	31% project cancelled

Cook, S.C. (2000). "What the Lessons from Large, Complex, Technical Projects Tell Us about the Art of Systems Engineering". INCOSE Symposium, Minneapolis.

Additional Facts Validate this State of Practice in the IT Sector

Some additional facts about the information technology sector…

- In the United States annually
  - Approximately 175,000 IT projects
  - Total cost exceeds $250 billion
  - Cost based on company size:
    - $2.322 million (large)
    - $1.331 million and (medium)
    - $ .434 million (small)
  - Only 16.2% completed on time within budget

  (Based on 1995 figures compiled by The Standish Group.)

- More research results on software projects
  - 31.1% are canceled before completion
  - 52.7% cost 189% of original estimate
  - Opportunity cost excluded ($1.1 M/day for Denver Airport)
  - Large company success lower (9.2%); deployed projects possess 42% of original functions
  - 48% of IT executives sampled believe failures > 5

Even more (recent) facts…

- A more recent (bleaker?) picture*
  - NIST estimates that software “bugs” cost American companies $60B in 2001
  - Guttman of CMU pegs that figure as low by a factor of 3 or 4
  - Calls for fundamental changes
    - Abandon “pre-industrial” Cobol and C
    - Emphasize integration, testing in education

* “Battling the Bugs,” *Financial Times*, London, 27 August 2002

Some underlying causes include…

- **Inadequate budget or too little time**
- **Poor planning**
- **Continually changing goals**
- **No foundations in software disciplines**
- **Lack of knowledge of advanced technology**
- **Insufficient oversight and communications**

And these are the reasons why IT architects and engineers believe they are unable to apply SE principles and concepts.

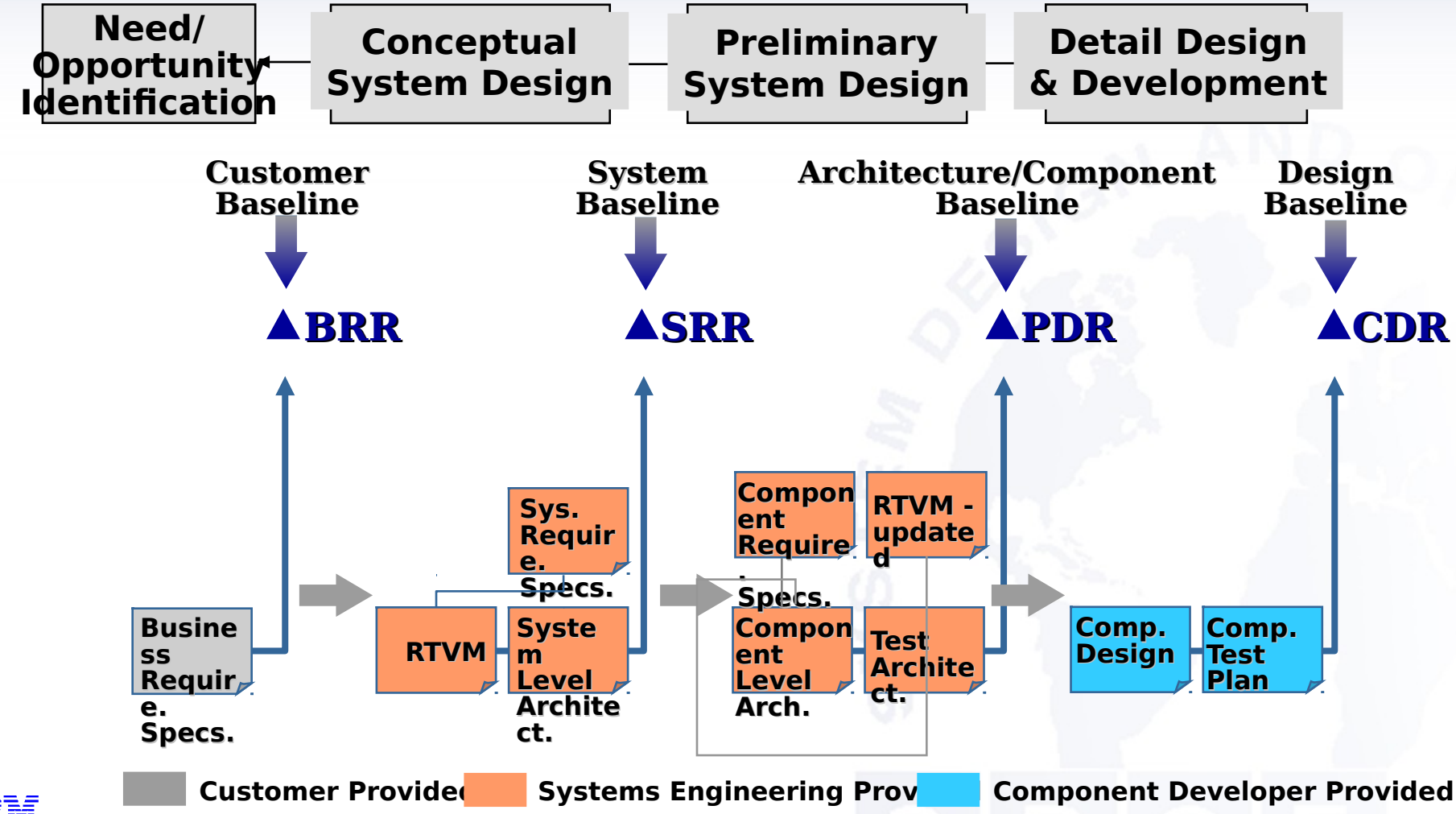
- Customer requirements and (even) identity (of customer) not clear
- Undocumented system scope and functionality
- Can't freeze the baseline
- Too many requirements – Maybe we can just do the key ones
- Global scope – multiple business processes with multiple owners
- Isolation from real "user"
- Executive management doesn't buy in
- Lack of teamwork
- Program Managers not empowered

Recurrent Themes: Ambiguous Requirements and Project Scope, Multiple and Often Conflicting Processes, Unclear Accountability.

Presentation Outline

- Relevance of Systems Engineering to the IT Industry
  - Forensics of a project close to home…
- Adapted Systems Engineering Process for the IT Industry – The IBM Approach
  - Baselines and Reviews
  - Sample Templates
- Cost of Systems Engineering
  - Sample Staffing Structures
- Benefits of Systems Engineering
  - Difficulties with metrics
- Concluding Remarks

A clearer correlation of SEA deliverables with the various defined milestones and design reviews…



System Requirements Review (SRR) Template

Contents:

1. **Template Development History**
2. **Goals and Objectives**
3. **Ground-rules**
4. **Entry and Exit Criteria**
5. **System Requirements Categories**
6. **Requirements Traceability and Verification Matrix**
7. **SRR Scoring Mechanism**
8. **SRR Scorecard**
9. **SRR Sample Agenda**
10. **SRR Signoffs**

Version: 2.0a (April 30, 2002)

System Requirements Review (SRR) Template:

Goal and Objectives

- GOAL: Convey a clear understanding of the business/stakeholder needs, rationale and priorities, and review the system level solution requirements and the IT solution approach - Get customer concurrence on system requirements
- Review and approve the documented System Requirements
  - Ensure that agreed to system requirements are unambiguous and testable
- Establish Traceability
  - The system requirements must be traceable upwards to the stakeholder requirements and business process requirements and downwards to acceptance criteria
- Establish the Technical Baseline
  - The system requirements represent the solution requirements baseline for a project
- Identify Technical Risks
  - Technology and Standards; Requirements and Acceptance Criteria Ambiguity; Technical Skill and Capability Requirements; Technical Approach Impact on Cost and Schedule
- Review mitigation plans
  - Ensure that the implementation approach selected addresses the deployment of the solution being developed, together with impact on the existing platforms and business processes
- Identify Dependencies (External - Technology, Baselines, Interfaces)
- Establish plans
  - Test Approach
  - System Requirements Baseline Created at SRR, further changes must follow change control process outlined in the Program Management Plan
- Identify Technical Performance Measures (TPMs)

System Requirements Review (SRR) Template:
Sample Agenda:

Topic	Time Slot

- SRR Objectives and Exit Criteria
- Business Process Definition
  - Problem and Solution Scope (Reference Baseline)
  - Significant Use Case Scenarios
  - Business Requirements and Priorities
- System Requirements Definition
  - Major Requirements Categories
  - Requirements Tradeoffs
  - Implementation/Technology Tradeoffs
- System Requirements Traceability
- Acceptance Criteria
- Project Plans
- Scoring
- Signoffs; Issues and Actions Summary

Presentation Outline

- **Relevance of Systems Engineering to the IT Industry**
  - **Forensics of a project close to home…**
- **Adapted Systems Engineering Process for the IT Industry – The IBM Approach**
  - **Baselines and Reviews**
  - **Sample Templates**
- **Cost of Systems Engineering**
  - **Sample Staffing Structures**
- **Benefits of Systems Engineering**
  - **Difficulties with metrics**
- **Concluding Remarks**

Presentation Outline

- Relevance of Systems Engineering to the IT Industry
  - Forensics of a project close to home…
- Adapted Systems Engineering Process for the IT Industry – The IBM Approach
  - Baselines and Reviews
  - Sample Templates
- Cost of Systems Engineering
  - Sample Staffing Structures
- Benefits of Systems Engineering
  - Difficulties with metrics
- Concluding Remarks

Benefits of Systems Engineering –

Difficulties “proving” benefits of systems engineering

- Since the project is not done twice with and without SE there is no way to know where the project fit into the statistics shown at the start of this presentation
  - **Project completes and is uneventful – e.g. meets the plan**
- Need a comprehensive data base of projects with the proper metrics collected to demonstrate the benefit
- Initial demonstration is subjective

- Demonstrate with standard project metrics
  - Function point, source lines of code or complexity estimates
  - Defects per function point or sloc by phase
  - Calculation of cost and schedule benefits
- Scoring of reviews results as green, yellow or red with respect to criteria to provide “in process” metrics

Benefits of Systems Engineering – One Study

For each software application, an SE assesses application complexity and application impact.

Application complexity is an integer from 0-8. It is determined by totaling applicable characteristics below...

1. Advanced exception processing
2. Has middleware interfaces
3. Has a GUI
4. Involves complex algorithms
5. Requires real-time response
6. Within the critical path
7. Involves batch processing
8. Involves data management

Application impact is a numeric value: 1, 20, 60, or 100. Qualitative equivalents listed respectively are none, minor, major, and new.

Assign a value of none (1) to application impact when the application exists and will not change--it resides in the E2E environment and will be involved in the SIT.

Assign a value of major (60) to application impact when the change involves...

adding or changing an interface; adding or changing functionality; adding or altering a middleware interface or interaction between online and batch processing; changing or updating the underlying infrastructure or middleware (for example, DB2 or operating system);

Benefits of Systems Engineering – One study

Launch (Project)	#of Points	Cost ($K)	$ / Point	Use SE?
System 1	12,934	30,000	2,319	No
System 2	10,209	14,904	1,460	Yes
System 3	4,678	6,614	1,414	Yes
System 4	8,707	18,075	2,076	No
System 5	1,223	2,400	1,962	No
System 5	4,600	10,309	2,241	Yes
Total/Average	**42,351**	**82,302**	**1943**	**N/A**
Total/Average with SE	**19,487**	**31,827**	**1,633**	**Yes**
Total/Average without SE	**22,864**	**50,475**	**2,208**	**No**
Percent improvement			**35.17%**	

Over a two year span, IBM has seen a 35% cost saving (productivity improvement) in large-scale integration projects that use the Systems Engineering process.

Presentation Outline

- Relevance of Systems Engineering to the IT Industry
  - Forensics of a project close to home…
- Adapted Systems Engineering Process for the IT Industry – The IBM Approach
  - Baselines and Reviews
  - Sample Templates
- Cost of Systems Engineering
  - Sample Work Breakdown Structures
- Benefits of Systems Engineering
  - Difficulties with metrics
- Concluding Remarks

In Conclusion…

- **Systems Engineering and Architecture Implementation**
  - The process must be "productized" for efficient implementation
    - Globally consistent templates, processes, tools and training
    - Uniform and consistent metrics and lexicon (part of the SE culture)
    - Consistent tailoring for various implementation approaches (structured, OO, iterative, …
    - Tailoring for time-to-market considerations,
    - Tailoring for schedule and resource considerations,
    - Risk tolerance must be explicitly considered in the tailoring process
  - Focus must be on the "necessary" and critical subset of the overall methodology and theory
  - Implementation must be organizationally supported and nurtured
    - Linkage to strategic organizational goals is key
    - Focused pilots on small projects help with process mechanics

Our Perspective…

- **Most SE Deployment and Implementation Efforts begin and end with Handbook and Guides on Systems Engineering!**
  - **And the cycle repeats every 4-6 years…**
- **It is absolutely key that "business drivers and the strategic intent for implementing SE be clearly delineated," thereafter, some initiatives are critical:**
  - **Practice of systems engineering needs a process, templates, tools, examples, case studies, metrics and supporting education. SE principles must be captured at various levels to convey valued to engineers, project managers, customers and executives**
  - **SE must focused on project schedules and costs… and the next milestone.**